

## **The Software Engineering Code of Ethics and Professional Practice**

<https://ethics.acm.org/code-of-ethics/software-engineering-code/>

<https://www.acm.org/binaries/content/assets/code-of-ethics/se-code-spn.pdf>

### **¿Por qué es esencial que los ingenieros de software prioricen el bienestar público?**

Debido a que los sistemas de software juegan un papel central en campos de la vida diaria como el comercio, industria, gobierno, medicina, educación, entretenimiento, y sociedad, por lo que cualquier falla o sesgo que se presente puede tener consecuencias significativas sobre la seguridad y calidad de vida de las personas.

Porque los ingenieros de software tienen la capacidad de diseñar y construir herramientas que entran a formar parte del día a día de las personas, a través del determinante entorno digital actual, que cada vez está más entrelazado con el estilo de vida y el desarrollo de las actividades humanas en una amplia variedad de contextos. Así, el hecho de ignorar las influencias personales, posibilidades de uso y las diversas consecuencias que el producto y su mantenimiento conllevan constituye una irresponsabilidad, debido a la posición profesional que se ejerce. De modo que corresponde a los ingenieros de software brindar productos y servicios en concordancia con el cliente y empleador, sin perder nunca de vista la primacía del bienestar público.

### **¿Qué significa mantener la calidad del producto en software?**

Significa que el software debe ser adaptable a nuevas situaciones a medida que estas vayan ocurriendo, asegurar que estemos calificados en cualquier proyecto en el que vayamos a trabajar, mantener la integridad de los datos, asegurarnos de que la documentación sea adecuada.

Asegurarse de que todo documento relacionado al producto haya sido debidamente aprobado; esforzarse por desarrollar el mejor código posible desde los conocimientos, estimar un costo aceptable, establecer un horario y compensaciones consideradas por las partes implicadas (trabajadores, clientes y empleadores); abordar las cuestiones éticas (sociales, medioambientales) y legales ligadas a los proyectos; asegurarse de usar métodos y estándares apropiados para cada proyecto, cumpliendo con los requerimientos y documentación que incluya problemas y soluciones encontrados, así como cumpliendo con la aprobación y revisión (pruebas y depuración); respetar la privacidad y usar, éticamente, datos precisos; finalmente, realizar el mantenimiento con el mismo profesionalismo del desarrollo.

### **¿Qué implica el compromiso con el aprendizaje continuo?**

Toda la vida debemos mejorar nuestras habilidades de documentación y producción de software, la comprensión de software de otros, el conocimiento de los estándares actuales.

Implica ampliar constantemente los conocimientos técnicos sobre el análisis, diseño y desarrollo de software, incluyendo la documentación; también, encaminarse a mejorar la confiabilidad, rentabilidad y eficiencia del software y de su proceso de desarrollo. Finalmente, no perder de vista los estándares legales y la buena ética.

### **¿Qué prácticas promueven la transparencia y la integridad en el desarrollo de software?**

Ser honestos en cuanto a las limitaciones de nuestra experiencia y educación a la hora de desarrollar software, no usar software obtenido mediante medios ilegales o no éticos, respetar la privacidad de aquellos a quienes está dirigido el software, usar datos obtenidos sólo por medios éticos y legales.

Aceptar total responsabilidad del propio trabajo, ejercer el juicio propio priorizando los valores humanos y el interés público, avalar documentos solamente si se tiene la competencia requerida y se está de acuerdo, brindar información financiera y técnica realista, dar a conocer cualquier conflicto de interés presentado. En caso de dirigir un proyecto, brindar un ambiente que promueva la calidad y bienestar para el desarrollo, informando oportunamente los estándares y políticas, teniendo en cuenta además las habilidades individuales; también, dar un trato justo en la evaluación, reconocimiento y remuneración de los desarrolladores o ingenieros.

### **¿Por qué es importante el respeto y apoyo entre colegas en el campo de la ingeniería de software?**

Un buen ambiente organizacional promueve el actuar éticamente, reportar violaciones al código de ética, reconocer los méritos del trabajo de otros, no intervenir injustamente en la carrera de alguien más, revisar el trabajo de otros en forma objetiva, sincera y propiamente documentada.

La importancia de esta adecuada cooperación reside en el trabajo en equipo ligado a la ingeniería de software y lo relevantes que resultan por ello las metodologías de trabajo; la transparencia y comunicación asertiva conllevan un proceso eficiente de desarrollo conjunto. Así, prácticas concretas como las siguientes destacan la influencia del respeto y el apoyo también frente a áreas transversales a la ingeniería de software, como la ética, la educación y la trayectoria profesional: contribuir a un buen ambiente organizacional, extender el conocimiento de la ingeniería de software y su ética, asistir a los colegas en dificultades técnicas o profesionales, dar crédito a y revisar el trabajo de otros objetivamente, escuchar las opiniones encontradas y no juzgar ni intervenir injustamente los procesos de los colegas.

### **¿Por qué es esencial que los ingenieros de software prioricen el bienestar público?**

Los ingenieros de software juegan un papel activo en la sociedad actual. El software que desarrollan impacta en la vida de las personas en el modo en cómo estas pueden relacionarse con la tecnología y especialmente con el software. Además, el software puede llegar a un número casi ilimitado de personas debido a que este se puede aplicar en ambientes multidisciplinarios en el campo no solo laboral sino educativo, por lo cual, muchas personas se pueden beneficiar o perjudicar por el software que ellos utilicen.

### **¿Qué significa mantener la calidad del producto en software?**

Mantener la calidad de un producto de software implica respetar ciertos estándares, y requisitos establecidos por el cliente que esté comprando el software. Fuera de las peticiones del cliente, hay ciertos estándares y buenas prácticas éticas y profesionales que se deben cumplir para mantener una buena calidad en el software, como establecer precios justos, calendarios y fechas de entrega fijas, registros, comunicación activa con el cliente durante el proceso de desarrollo, etc. Todas estas prácticas hacen que la calidad del producto final sea más acorde a lo exigido durante el trabajo, y reduce la probabilidad de problemáticas con el cliente durante el desarrollo. Esto último es sumamente importante ya que al fin y al cabo, se tiene que cumplir con un software que cumpla de forma real las expectativas del cliente.

### **¿Qué implica el compromiso con el aprendizaje continuo?**

Implica estar actualizado al desarrollo de las tecnologías actuales, no solo por el tema de mantener el software que uno desarrolle a un nivel estándar para la época en la que se desarrolla (garantizando una mejor calidad al cliente), sino también para tener un mejor entendimiento del software que desarrollen otros. Por último, hay que tener muy presente lo relacionado a todo el marco legal y social, ya que esto puede tener un MUY fuerte impacto en lo relacionado a todo lo que es la vida laboral de uno mismo como desarrollador de software, incluso llegando a peligrar su estabilidad.

### **¿Qué prácticas promueven la transparencia y la integridad en el desarrollo de software?**

Algunas prácticas son:

- Protección de datos del (los) usuario(s).
- Exactitud respecto a fechas de entrega.
- No usar software ilegalmente o de agentes externos sin consentimiento.
- No realizar software malicioso.
- Valuación correcta del trabajo. Sin menospreciar el trabajo de uno, pero tampoco exigir precios exorbitantes.

### **¿Por qué es importante el respeto y apoyo entre colegas en el campo de la ingeniería de software?**

El respeto es crucial en el ámbito de desarrollo y diseño de productos en general, ya que la falta de respeto en el ambiente laboral, puede traducirse en robo del trabajo de otros. Esto tiene como consecuencia el uso ilegal de software ajeno en proyectos con fines lucrativos, quitando dinero y reconocimiento al desarrollador original. Además, el apoyo entre colegas promueve las denuncias ante estos actos de uso ilegal de software, o propagación de software malicioso que perjudique al consumidor y a otros desarrolladores.